

**IN THE UNITED STATES PATENT AND TRADEMARK OFFICE**

In re Application of:

NAKAI et al.

Application No.: Unassigned

Art Unit: Unassigned

Filed: May 1, 2001

Examiner: Unassigned

For: A SYSTEM FOR  
DEVELOPING AN  
APPLICATION SYSTEM  
AND IMPLEMENTING  
THEREOF

**PRELIMINARY AMENDMENT**

Commissioner for Patents  
Washington, D. C. 20231

Dear Sir:

Prior to the examination of the above-identified patent application, please enter the following amendments and consider the following remarks.

**IN THE TITLE**

Replace the title with: A SYSTEM FOR DEVELOPING AN APPLICATION  
SYSTEM AND IMPLEMENTATION THEREOF.

**IN THE SPECIFICATION**

Replace the paragraph beginning at page 1, line 6, with:

The invention relates to a system for developing an application system and  
implementation thereof for an automated machine.

Replace the paragraph beginning at page 1, line 10, with:

Various types of controllers such as a programmable logic controller (PLC) and a motion controller have been utilized to configure well-known automated machines including, for example, machine tools in the art of factory automation, industrial automated machines such as industrial robots, self-controlled robots, and typical automated machines. Further, the controller is connected with various types of input devices including a sensor and a switch for inputting signals thereto, also with many kinds of output devices including a motor and a display to be controlled thereby. Also, a program in the form of software describing commands (instructions) for the input and output devices (i.e., I/O devices) is installed in the controller. The automated machines are operated upon receiving the commands described in the software program.

Replace the paragraph beginning at page 1, line 25, with:

As indicated above, the input and output devices are connected to Input/Output connectors (i.e., I/O ports) of the controller. In case where the conventional controller is utilized, first of all, a programmer of a controller needs to precisely recognize which of the I/O ports is connected with which one of I/O devices, and then sets up software modules or operation programs for I/O devices implemented within the controller. Otherwise, the controller can not send appropriate commands to the I/O devices so that the controller fails to control the I/O devices. Such software modules include, for example, a device driver.

Replace the paragraph beginning at page 2, line 12, with:

The aforementioned disadvantage has not been solved up to the last couple of years, in which an expansion card and a peripheral device are connected with a conventional PC (personal computer). Such a peripheral device is also referred to as an object to be controlled or simply as an object. The user must exactly understand the connections between I/O ports located on the PC and peripheral devices, then, the user

may properly install software modules and/or operation programs for the I/O devices to be executed with the PC.

Replace the paragraph beginning at page 2, line 22, with:

Meanwhile, another type of the interface has recently been developed with a concept of "Plug and Play (PnP)" for connection between the PC and the peripheral devices. The above-mentioned PnP interface is, for example, "USB (Universal Serial Bus)" or "IEEE1394 (Institute of Electrical and Electronics Engineers 1394)" standards. In fact, use of the PnP interface reduces a burden of the user. Each of peripheral devices having the PnP interface is assigned a unique identification, named "GUID (Global Unique ID)".

Replace the paragraph beginning at page 3, line 7, with:

As described above, the GUID identification is globally unique and individually assigned to each object. Suppose that the PC has already stored a predetermined number of software modules such as device drivers for driving corresponding objects. Once the I/O object is connected with the PC, the PC acquires the GUID of the I/O object. Then, the PC automatically identifies the corresponding software module according to the GUID so as to drive the I/O device. Therefore, this eliminates the need for the user to select and set up a suitable software module such as a device driver. To this end, the user may not even be aware of the fact that the software module is installed within the PC, while the user can connect with the I/O device to utilize it. Yet, even though the user has to input and store the option parameters of the software module, this task may also be avoided if the predetermined initial values thereof are used.

Replace the paragraph beginning at page 3, line 24, with:

Also, the standards of USB and IEEE1394 have another function based upon the concept of "Hot Plug". The conventional PC is required to shut down and be rebooted to activate a software module that is newly installed for an additional device. The Hot Plug enables the user to connect another device to the PC without shutdown so that almost no task is required for connecting the device with the PC. Thus, the user undertakes much less burden than that in utilizing the conventional PC.

Replace the paragraph beginning at page 4, line 9, with:

The PnP interface such as the USB and/or IEEE1394 standards has another feature. That is, the communication protocol of the PnP interface is open to the public. In other words, the communication protocol thereof is public. This allows various third parties to develop peripheral devices for the PC. Connection of the conventional PC with a peripheral device requires an expansion board, such as a particular connecting board, for connecting therebetween, however, the feature of publicity eliminates this trouble.

Replace the paragraph beginning at page 4, line 19, with:

The aforementioned feature of publicity is realized in the communication protocol between the PC and the peripheral device. Besides, demand has been increased so that the feature of publicity is realized on the communication protocol between the controller and the peripheral device. Various controllers have been proposed such that they include interfaces such as the USB and/or IEEE1394 standards with the feature of the publicity.

Replace the paragraph beginning at page 5, line 2, with:

As described above, the recent controllers and PCs are going to support the PnP interface such as USB and/or IEEE1394 standards. However, the feature of the Plug and Play has not yet been exploited effectively enough to utilize the device instantly after connecting it to the PC or the controllers.

Replace the paragraph beginning at page 5, line 8, with:

Firstly, the operation of the peripheral device (object) controlled by the PC is described hereinafter. When the user, for example, moves a typical mouse as a peripheral device, a mouse cursor moves on the monitor in accordance with the actual movement of the mouse. Thus, the operation of the mouse cursor is preprogrammed and unmodified with the mouse and the PC. A USB camera is another example as the peripheral device, in which a particular software program for displaying images, taken by the USB camera, on the PC monitor is distributed with the USB camera by the manufacturer thereof. The USB camera cannot be operated without using the software program. To this end, the operation manner of the peripheral devices and the operation environment of the PC are fixed and cannot be modified by the user (through instruction with program codes).

Replace the paragraph beginning at page 6, line 8, with:

Furthermore, even if the controller is adapted to use the interfaces such as USB or IEEE1394 standards, the controller has to be installed with the software module or the operation program implementing it, precisely as on the conventional controllers. Therefore, the advantage of the "Plug and Play" feature cannot be sufficiently exploited in those circumstances.

Replace the paragraph beginning at page 6, line 15, with:

The present invention is intended for use of a system for configuring a automated machine comprising peripheral devices (object) and controllers with PnP interfaces (i.e., having the feature of Plug and Play).

Replace the paragraph beginning at page 13, line 2, with:

Figs. 4A through 4C are block diagrams showing another software architecture including a plurality of peripheral devices according to Embodiment 1 of the present invention;

#### IN THE CLAIMS

Replace the existing claims with:

1. (Amended) A system comprising:
  - a controller;
  - a monitor connected with said controller;
  - at least one object to be controlled, said object being connected to said controller;
  - development means for developing a program for said object;
  - implementing means for implementing the program developed by said development means; and
  - a software module uniquely assigned to said object and providing at least one of
    - an icon procedure for displaying an icon for said object in a display area on said monitor,
    - a description procedure for describing a control process for said object, and
    - an implementing procedure for implementing the control process developed for said object.

2. (Amended) The system according to Claim 1, wherein said object includes at least one device from which said development means acquires a global unique ID, and said development means identifies said software module with the global unique ID.

3. (Amended) The system according to Claim 2, wherein said software module is stored within said object so that said development means acquires said software module from said controlled object.

4. (Amended) The system according to Claim 2, wherein said software module is stored within a database server connected with said development means through a communication bus so that said development means acquires said software module from the database server.

5. (Amended) The system according to Claim 2, wherein said development means provides a display area on the monitor in which at least one icon is displayed, the icon representing one of said object connected to said controller and an object to be connected to said controller.

6. (Amended) The system according to Claim 5, wherein the icon procedure displays a plurality of icons in the display area on said monitor, each icon illustrating current status of said object.

7. (Amended) The system according to Claim 5, wherein said development means provides a development area on said monitor, and a user copies the icon from the display area onto the development area, thereby developing the program.

8. (Amended) The system according to Claim 7, wherein the user utilizes the description procedure for describing a control process for said object determining operation of said object, thereby developing the program.

9. (Amended) The system according to Claim 8, wherein the icon procedure displays a plurality of icons in the display area on said monitor, each icon illustrating operation of said object.

10. (Amended) The system according to Claim 7, wherein the user connects a plurality of the icons with each other to form a flowchart in the development area, thereby developing the program.

11. (Amended) The system according to Claim 9, wherein said development means displays the icons in the display area, and simulates operation of said object while execution of the program is simulated, whereby the monitor is used for displaying simulation of said object.

12. (Amended) The system according to Claim 6, wherein said development means displays the icons in the display area, illustrates operation of said object while said implementing means implements the program, whereby the monitor is used for displaying operation of said object.

13. (Amended) The system according to Claim 7, wherein said implement means sends messages to and/or receives messages from said object according to the program developed.

14. (Amended) The system according to Claim 7, wherein said object is connected to said controller through an interface including at least one of a Plug and Play function and a Hot Plug function.



15. (Amended) A storage medium storing a computer program for execution on a system which comprises

a controller,

a monitor connected to said controller,

at least one object to be controlled, said object being connected to said controller,

development means for developing a program for said controlled object,

implementing means for implementing the program developed by said

development means, and

a software module uniquely assigned to said object, said software module including an icon procedure for displaying an icon for said object in a display area on said monitor, a description procedure for describing a control process for said object, and an implementing procedure for implementing the control process developed for said object,

said system including said object including at least one device,

said computer program comprising:

a first subprocess in which said development means acquires a global unique ID from said device;

a second subprocess in which said development means identifies said software module with the global unique ID;

a third subprocess in which said development means provides a display area on the monitor, in which at least one icon is displayed, the icon representing one of said object connected to said controller and an object to be connected to said controller;

a fourth subprocess in which said development means provides a development area on said monitor; and

a fifth subprocess in which the icon is copied from the display area onto the development area, thereby developing an application program.

IN THE ABSTRACT

Replace the abstract with:

ABSTRACT OF THE DISCLOSURE

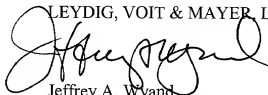
A system for developing application systems and implementing the system for an automated machine, which can reduce the burden of installing the peripheral devices with a controller. The system includes a controller, a monitor connected with the controller, and at least one object to be controlled. The object is connected to the controller. The control system further includes developing a program for the controlled object, implementing the program developed, and a software module uniquely assigned to the object. The software module provides at least one of an icon procedure for displaying an icon for the object in a display area on the monitor, a description procedure for describing a control process for the object, and an implement procedure for implementing the control process developed for the object.

**REMARKS**

The foregoing amendments are made to correct minor translational errors and to meet United States requirements as to form. No new matter is added.

Respectfully submitted,

LEYDIG, VOIT & MAYER, LTD.



Jeffrey A. Wyand,  
Registration No. 29,458

Suite 300  
700 Thirteenth Street, N. W.  
Washington, D. C. 20005  
Telephone: (202) 737-6770  
Facsimile: (202) 737-6776  
Date: May 1, 2001  
JAW:cmcg

**IN THE UNITED STATES PATENT AND TRADEMARK OFFICE**

In re Application of:

NAKAI et al.

Application No.: Unassigned

Art Unit: Unassigned

Filed: May 1, 2001

Examiner: Unassigned

For: A SYSTEM FOR  
DEVELOPING AN  
APPLICATION  
SYSTEM AND  
IMPLEMENTING  
THEREOF

**SPECIFICATION, CLAIMS AND  
ABSTRACT AS PRELIMINARILY AMENDED**

Amendments to the paragraph beginning at page 1, line 6:

The invention relates to a system for developing an application system and ~~implementing~~implementation thereof for an automated machine.

Amendments to the paragraph beginning at page 1, line 10:

Various types of controllers such as a programmable logic controller (PLC) and a motion controller have been utilized to configure well-known automated machines including, for example, machine tools in the art of factory automation, industrial automated machines such as industrial robots, self-controlled robots, and typical automated machines. Further, the controller is connected with various types of input devices including a sensor and a switch for inputting signals thereto, also with many kinds of output devices including a motor and a display to be controlled thereby. Also, a program in ~~a~~ the form of ~~a~~ software describing commands (instructions) for the input and

output devices (i.e., I/O devices) is installed in the controller. The automated machines are operated upon receiving the commands described in the software program.

Amendments to the paragraph beginning at page 1, line 25:

As indicated above, the input and output devices are connected to Input/Output connectors (i.e., I/O ports) of the controller. In case where the conventional controller is utilized, first of all, a programmer of a controller needs to precisely recognize ~~that each~~ which of the I/O ports is connected with which one of I/O devices, and then sets up software modules or operation programs for I/O devices ~~implementing implemented~~ within the controller. Otherwise, the controller can not send appropriate commands to the I/O devices so that the controller fails to control the I/O devices. Such software modules include, for example, a device driver.

Amendments to the paragraph beginning at page 2, line 12:

The aforementioned disadvantage has not been solved up to ~~a the~~ the last couple of years, in which an expansion card and a peripheral device are connected with ~~the a~~ conventional PC (personal computer). Such a peripheral device is also referred to as an object to be controlled or simply as an object. The user must exactly understand the connections between I/O ports located on the PC and peripheral devices, then, the user may properly install software modules and/or operation programs for the I/O devices to be executed with the PC.

Amendments to the paragraph beginning at page 2, line 22:

Meanwhile, ~~an~~ another type of the interface has recently been developed with a concept of "Plug and Play (PnP)" for connection between the PC and the peripheral devices. The above-mentioned PnP interface is, for example, "USB (Universal Serial Bus)" or "IEEE1394 (Institute of Electrical and Electronics Engineers 1394)" standards. In fact, use of the PnP interface reduces a burden of the user. Each of peripheral devices

having the PnP interface is ~~uniquely assigned with a~~ unique identification, named "GUID (Global Unique ID)".

Amendments to the paragraph beginning at page 3, line 7:

As described above, the GUID ~~is the identification, which~~ is globally unique, and individually assigned to each object. Suppose that the PC has already stored a predetermined number of software modules such as device drivers for driving corresponding objects. Once the I/O object is connected with the PC, the PC acquires the GUID of the I/O object. Then, the PC automatically identifies the corresponding software module according to the GUID so as to drive the I/O device. Therefore, this eliminates the need for the user to select and set up a suitable software module such as a device driver. To this end, the user may not even be aware of the fact that the software module is installed within the PC, while the user can connect with the I/O device to utilize it. Yet, even though the user has to input and store the option parameters of the software module, this task may also be avoided if the predetermined initial values thereof are used.

Amendments to the paragraph beginning at page 3, line 24:

Also, the standards of USB and IEEE1394 have another function based upon the concept of "Hot Plug". The conventional PC is required to shut down and ~~then be~~ rebooted ~~so as to activate the~~ a software module that is newly installed for an additional device. The ~~function of the~~ Hot Plug enables the user to connect another device to the PC without shutdown so that almost no task is required for connecting the device with the PC. Thus, the user undertakes ~~the burden~~ much less burden than that in utilizing the conventional PC.

Amendments to the paragraph beginning at page 4, line 9:

The PnP interface such as the USB and/or IEEE1394 standards has another feature. That is, the communication protocol of the PnP interface is open to the public. In other words, the communication protocol thereof ~~has publicity~~ is public. This allows various third parties to develop peripheral devices for the PC. Connection of the conventional PC with a peripheral device requires an expansion board, such as a particular connecting board, for connecting therebetween, however, the feature of ~~the publicity eliminates the troublesome~~ this trouble.

Amendments to the paragraph beginning at page 4, line 19:

The aforementioned feature of ~~the publicity~~ is realized on in the communication protocol between the PC and the peripheral device. Besides, ~~a~~ demand has been increased so that the feature of ~~the publicity~~ is realized on the communication protocol between the controller and the peripheral device. Various controllers have been proposed such that they ~~equip~~ include interfaces such as the USB and/or IEEE1394 standards with the feature of the publicity.

Amendments to the paragraph beginning at page 5, line 2:

As described above, the recent controllers and PCs are going to support the PnP interface such as USB and/or IEEE1394 standards. However, the feature of the Plug and Play has not yet been exploited effectively enough to utilize the device instantly after connecting ~~thereof with it~~ to the PC or the controllers.

Amendments to the paragraph beginning at page 5, line 8:

Firstly, the operation of the peripheral device (object) controlled by the PC is described hereinafter. When the user, for example, moves a typical mouse as ~~one of the a~~ peripheral device, a mouse cursor moves on the monitor in accordance with the actual

movement of the mouse. Thus, the operation of the mouse cursor is preprogrammed and unmodified with the mouse and the PC. A USB camera is another example as the peripheral device, in which a particular software program for displaying images, taken by the USB camera, on the PC monitor is distributed with the USB camera by the manufacturer thereof. The USB camera cannot be operated without using the software program. To this end, the operation manner of the peripheral devices and the operation environment of the PC are fixed and cannot be modified by the user (through instruction with program codes).

Amendments to the paragraph beginning at page 6, line 8:

Furthermore, even if the controller is adapted to use the interfaces such as USB or IEEE1394 standards, the controller has to be installed with the software module or the operation program implementing ~~thereon it, as~~ precisely as on the conventional controllers. Therefore, the advantage of the "Plug and Play" feature cannot be sufficiently exploited in those circumstances.

Amendments to the paragraph beginning at page 6, line 15:

The present invention is intended for use of a system for configuring ~~the a~~ automated machine comprising peripheral devices (object) and controllers with ~~the~~ PnP interfaces (i.e., having the feature of Plug and Play).

Amendments to the paragraph beginning at page 13, line 2:

Figs. 4A through 4C are block diagrams showing another software ~~architectures~~ architecture including a plurality of peripheral devices according to Embodiment 1 of the present invention;



Amendments to the existing claims:

1. (Amended) A system comprising:  
a controller;  
a monitor connected with said controller;  
at least one object to be controlled, said object being connected ~~with~~ to said controller;  
development means for developing a program for said ~~controlled~~ object;  
~~implement~~ implementing means for implementing the program developed by said development means; and  
a software module uniquely assigned to said object, ~~said software module being in a form of software and~~ providing at least one of ~~procedures including~~,  
an icon procedure for displaying an icon for said object in a display area on said monitor,  
a description procedure for describing a control process for said object, and  
an ~~implement~~ implementing procedure for implementing the control process developed for said object.
2. (Amended) The system according to Claim 1, wherein  
said object includes at least one device from which said development means acquires a global unique ID ~~or another similar data thereto~~, and  
~~wherein~~ said development means identifies said software module with the global unique ID ~~or another similar data thereto~~.
3. (Amended) The system according to Claim 2, wherein said software module is stored within said object so that said development means acquires said software module from said controlled object.

4. (Amended) The system according to Claim 2, wherein said software module is stored within a database server connected with said development means through a communication bus so that said development means acquires said software module from the database server.

5. (Amended) The system according to Claim 2, wherein said development means provides a display area on the monitor, in which at least one icon is displayed, the icon representing ~~for one of~~ said object connected to said controller ~~or said and an~~ object to be connected to said controller.

6. (Amended) The system according to Claim 5, wherein the icon procedure displays a plurality of icons in the display area on said monitor, each icon illustrating a current status of said object.

7. (Amended) The system according to Claim 5, wherein said development means provides a development area on said monitor, and ~~wherein the~~ user copies the icon from the display area onto the development area, thereby ~~to develop~~ developing the program.

8. (Amended) The system according to Claim 7, wherein the user utilizes the description procedure for describing a control process for said object determining ~~an~~ operation of said object, thereby ~~to develop~~ developing the program.

9. (Amended) The system according to Claim 8, wherein the icon procedure displays a plurality of icons in the display area on said monitor, each icon illustrating ~~an~~ operation of said object.

10. (Amended) The system according to Claim 7, wherein the user connects a plurality of the icons with each other to form a flowchart in the development area, thereby ~~to develop~~ developing the program.

11. (Amended) The system according to Claim 9, wherein said development means displays the icons in the display area, and simulates ~~the~~ operation of said object while execution of the program is simulated, whereby the monitor is used for displaying ~~the simulation thereof of said object.~~

12. (Amended) The system according to Claim 6, wherein said development means displays the icons in the display area, illustrates ~~the~~ operation of said object while said ~~implement~~ implementing means implements the program, whereby the monitor is used for displaying ~~the operation thereof of said object.~~

13. (Amended) The system according to Claim 7, wherein said implement means sends messages to and/or receives messages from said object according to the program developed ~~program.~~

14. (Amended) The system according to Claim 7, wherein said object is connected ~~with to~~ said controller through an interface ~~serving functions~~ including at least one of a Plug and Play function ~~or and~~ a Hot Plug function.

15. (Amended) A storage ~~media~~ medium storing a computer program for execution on a system which comprises

a controller,

a monitor connected ~~with to~~ said controller,

at least one object to be controlled, said object being connected with to said controller,

development means for developing a program for said controlled object,

~~implement~~ implementing means for implementing the program developed by said development means, and

a software module uniquely assigned to said object, said software module including an icon procedure for displaying an icon for said object in a display area on said monitor, a description procedure for describing a control process for said object, and an

~~implement~~implementing procedure for implementing the control process developed for said object,

said system ~~in which including~~ said object ~~includes~~ including at least one device, said computer program comprising:

a first subprocess, in which said development means acquires a global unique ID ~~or another similar data thereto~~ from said device;

a second subprocess, in which said development means identifies said software module with the global unique ID ~~or another similar data thereto~~;

a third subprocess, in which said development means provides a display area on the monitor, in which at least one icon is displayed, the icon representing ~~for one of~~ said object connected to said controller ~~or said and an~~ object to be connected to said controller;

a ~~forth~~ fourth subprocess, in which said development means provides a development area on said monitor; and

a fifth subprocess, in which the icon is copied from the display area onto the development area, thereby ~~to develop~~ developing an application program.

Amendments to the abstract:

#### ABSTRACT OF THE DISCLOSURE

~~The present invention is to provide a~~ A system for developing application systems and implementing ~~thereof the system~~ for an automated machine, which can reduce the burden of ~~the user to install~~ installing the peripheral devices with a controller. The system ~~comprises~~ includes a controller, a monitor connected with the controller, and at least one object to be controlled. The object is connected ~~with~~ to the controller. The control system further ~~comprises development means for~~ includes developing a program for the controlled object, ~~implement means for implementing the program developed by the development means~~, and a software module uniquely assigned to the object. The software module ~~is in a form of software providing~~ provides at least one of ~~procedures including~~, an icon procedure for displaying an icon for the object in a display area on the

A10

**IN THE UNITED STATES PATENT AND TRADEMARK OFFICE**

In re Application of:

NAKAI et al.

Application No.: Unassigned

Art Unit: Unassigned

Filed: May 1, 2001

Examiner: Unassigned

For: A SYSTEM FOR  
DEVELOPING AN  
APPLICATION  
SYSTEM AND  
IMPLEMENTING  
THEREOF

**CLAIMS PENDING AFTER PRELIMINARY AMENDMENT**

1. A system comprising:  
a controller;  
a monitor connected with said controller;  
at least one object to be controlled, said object being connected to said controller;  
development means for developing a program for said object;  
implementing means for implementing the program developed by said  
development means; and  
a software module uniquely assigned to said object and providing at least one of  
an icon procedure for displaying an icon for said object in a display area on  
said monitor,  
a description procedure for describing a control process for said object, and  
an implementing procedure for implementing the control process  
developed for said object.
2. The system according to Claim 1, wherein  
said object includes at least one device from which said development means  
acquires a global unique ID, and

said development means identifies said software module with the global unique ID.

3. The system according to Claim 2, wherein said software module is stored within said object so that said development means acquires said software module from said controlled object.

4. The system according to Claim 2, wherein said software module is stored within a database server connected with said development means through a communication bus so that said development means acquires said software module from the database server.

5. The system according to Claim 2, wherein said development means provides a display area on the monitor in which at least one icon is displayed, the icon representing one of said object connected to said controller and an object to be connected to said controller.

6. The system according to Claim 5, wherein the icon procedure displays a plurality of icons in the display area on said monitor, each icon illustrating current status of said object.

7. The system according to Claim 5, wherein  
said development means provides a development area on said monitor, and  
a user copies the icon from the display area onto the development area, thereby  
developing the program.

8. The system according to Claim 7, wherein the user utilizes the description procedure for describing a control process for said object determining operation of said object, thereby developing the program.

9. The system according to Claim 8, wherein the icon procedure displays a plurality of icons in the display area on said monitor, each icon illustrating operation of said object.

10. The system according to Claim 7, wherein the user connects a plurality of the icons with each other to form a flowchart in the development area, thereby developing the program.

11. The system according to Claim 9, wherein said development means displays the icons in the display area, and simulates operation of said object while execution of the program is simulated, whereby the monitor is used for displaying simulation of said object.

12. The system according to Claim 6, wherein said development means displays the icons in the display area, illustrates operation of said object while said implementing means implements the program, whereby the monitor is used for displaying operation of said object.

13. The system according to Claim 7, wherein said implement means sends messages to and/or receives messages from said object according to the program developed.

14. The system according to Claim 7, wherein said object is connected to said controller through an interface including at least one of a Plug and Play function and a Hot Plug function.

15. A storage medium storing a computer program for execution on a system which comprises  
a controller,  
a monitor connected to said controller,  
at least one object to be controlled, said object being connected to said controller,



development means for developing a program for said controlled object,  
implementing means for implementing the program developed by said  
development means, and

a software module uniquely assigned to said object, said software module  
including an icon procedure for displaying an icon for said object in a display area on said  
monitor, a description procedure for describing a control process for said object, and an  
implementing procedure for implementing the control process developed for said object,  
said system including said object including at least one device,  
said computer program comprising:

a first subprocess in which said development means acquires a global unique ID  
from said device;

a second subprocess in which said development means identifies said software  
module with the global unique ID;

a third subprocess in which said development means provides a display area on the  
monitor, in which at least one icon is displayed, the icon representing one of said object  
connected to said controller and an object to be connected to said controller;

a fourth subprocess in which said development means provides a development  
area on said monitor; and

a fifth subprocess in which the icon is copied from the display area onto the  
development area, thereby developing an application program.